

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Минцаев Магомед Шавалович

Должность: Ректор

Дата подписания: 14.06.2024 10:15:23

Уникальный программный ключ:

236bcc35c296f119d6aafdc22836b21db52dbc07971a86865a5825f9fa4304cc

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ АКАДЕМИКА М.Д.МИЛЛИОНЩИКОВА»

Информационные системы в экономике

(наименование кафедры)

УТВЕРЖДЕН

на заседании кафедры
« 02 » 09 20__ г., протокол № 1

Заведующий кафедрой
Л.Р. Магомаева
(подпись)

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Разработка Web-приложений на языке JavaScript

(наименование дисциплины)

Направление /специальность подготовки

09.03.03 Прикладная информатика

(код и наименование направления/ специальности подготовки)

Направленность (профиль)

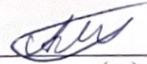
Разработка мобильных и Web-приложений

(наименование специализации / профиля подготовки)

Квалификация

бакалавр

(специалист / бакалавр / магистр)

Составитель  М.К. Абдулаев
(подпись)

Грозный – 2024

ПАСПОРТ
ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ
Разработка Web-приложений на языке JavaScript

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
5 семестр			
1	Тема 1. Введение в веб-разработку и JavaScript	(ОПК-2), (ПК – 5)	Лабораторная работа
2	Тема 2. Продвинутое концепции JavaScript	(ОПК-2), (ПК – 5)	Лабораторная работа
3	Тема 3. Работа с современными возможностями DOM	(ОПК-2), (ПК – 5)	Лабораторная работа
4	Тема 4. Практическое применение асинхронного программирования	(ОПК-2), (ПК – 5)	Лабораторная работа
5	Тема 5. Создание динамических интерфейсов с использованием шаблонизации	(ОПК-2), (ПК – 5)	Лабораторная работа
6	Тема 6. Разработка клиент-серверных веб-приложений с использованием Fetch API	(ОПК-2), (ПК – 5)	Лабораторная работа

ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
1	<i>Лабораторная работа</i>	Средство проверки умений применять полученные знания по заранее определенной методике для решения задач или заданий по модулю или дисциплине в целом	Комплект заданий для выполнения лабораторных работ
2	<i>Рубежный контроль</i>	Форма проверки знаний по дисциплине в виде первой и второй рубежных аттестаций	Вопросы к аттестациям
3	<i>экзамен</i>	Итоговая форма оценки знаний	Вопросы к экзамену

ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ

В качестве оценочных средств используются средства контроля выполнения лабораторных работ по дисциплине. Защита лабораторной работы – ответ на контрольные вопросы после выполнения лабораторной работы.

Средства текущего контроля: устный опрос (собеседование/опрос, разбор учебной ситуации на выбранную тему, подготовка устных сообщений и докладов), практическое задание (выполнение заданий в электронной форме на ПК).

Лабораторная работа 1.
Разработка базового веб-приложения

Цель лабораторной работы: Научиться создавать основную структуру HTML, добавлять базовое стилевое оформление с использованием CSS, инициализировать JavaScript файл и реализовывать базовый функционал для ввода данных и обработки событий.

(ОПК-2), (ПК-5)

Лабораторная работа 2.
Динамический интерфейс с использованием DOM

Цель лабораторной работы: Научиться использовать DOM API для динамического изменения содержимого страницы, создавать интерактивные элементы управления и обрабатывать пользовательские действия с помощью JavaScript.

(ОПК-2), (ПК-5)

Лабораторная работа 3.
Работа с серверными запросами

Цель лабораторной работы: Научиться настраивать серверную часть приложения для обработки запросов, использовать Fetch API для отправки асинхронных запросов на сервер и получения данных, а также обрабатывать ответы от сервера и динамически обновлять интерфейс.

(ОПК-2), (ПК-5)

Лабораторная работа 4.
Оптимизация и доработка приложения

Цель лабораторной работы: Научиться оптимизировать код JavaScript и CSS для улучшения производительности и читаемости, добавлять дополнительный функционал на основе полученных знаний, а также тестировать и отлаживать приложение.

(ОПК-2), (ПК-5)

Лабораторная работа 5.
Развертывание и
документирование

Цель лабораторной работы: Научиться развертывать приложение на хостинге или локальном сервере, создавать документацию для пользователя и разработчика, а также подготавливать проект к презентации и демонстрации.

(ОПК-2), (ПК-5)

Лабораторная работа 6.
Финальный проект

Цель лабораторной работы: Демонстрация приобретенных знаний и навыков путем самостоятельной разработки собственного веб-приложения, его презентации и обсуждения результатов в группе.

(ОПК-2), (ПК-5)

Критерии оценки ответов на лабораторные работы

Регламентом БРС предусмотрено всего 15 баллов за текущую работу студента.

Критерии оценки разработаны, исходя из возможности ответа студентом до 6 лабораторных работ с использованием дополнительного материала по ним. (по 2 балла).

2 балла ставится за работу, выполненную полностью без ошибок и недочетов.

1 балл ставится за работу, выполненную полностью, но при наличии в ней негрубых ошибок и недочетов.

0 баллов ставится, если студент совсем не выполнил ни одного задания.

Максимальное количество баллов за выполнение и защиту лабораторных работ 12.

Максимально высокие баллы за текущий контроль – 15. За активное участие на лекциях и использование дополнительного материала при подготовке к занятиям студент получает дополнительно до 3 баллов.

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ АКАДЕМИКА М.Д.МИЛЛИОНЩИКОВА**

**Институт цифровой экономики и технологического предпринимательства
Кафедра «Информационные системы в экономике»**

Вопросы к первой рубежной аттестации «Разработка Web-приложений на языке JavaScript»

1. Какова роль JavaScript в веб-разработке?
2. В чем заключаются основы синтаксиса JavaScript?
3. Что такое замыкания в JavaScript и как они используются?
4. Как работают прототипы в JavaScript?
5. Что такое асинхронное программирование и почему оно важно?
6. Какие методы позволяют работать с DOM-элементами?
7. Что такое делегирование событий и в чем его преимущества?
8. Какие типы событий существуют в JavaScript?
9. Что такое промисы и как они помогают в асинхронном программировании?
10. Как использовать `async/await` для управления асинхронными операциями?
11. Как осуществляется работа с API с использованием асинхронных запросов?
12. В чем заключается шаблонизация в веб-разработке?
13. Какие методы используются для создания динамических интерфейсов с помощью шаблонизации?
14. Какие возможности предоставляет Fetch API для обмена данными между клиентом и сервером?
15. Какие практические примеры использования Fetch API вы можете привести?

Вопросы ко второй рубежной аттестации «Разработка Web-приложений на языке JavaScript»

1. Каким образом JavaScript используется для динамического изменения содержимого веб-страницы?
2. Как работает создание, удаление и модификация DOM-элементов?
3. В чем отличие между синхронным и асинхронным выполнением кода в JavaScript?
4. Как можно организовать обработку событий в JavaScript?
5. Какие основные принципы лежат в основе работы с промисами?
6. Какие методы использования асинхронных запросов к API вы знаете?
7. Как можно оптимизировать работу с клиент-серверными веб-приложениями?
8. Какие методы и технологии используются для упрощения работы с интерфейсом в JavaScript?

9. Какие существуют подходы к шаблонизации в веб-разработке?
10. Какие основные принципы стоит учитывать при разработке динамических интерфейсов?
11. Какие проблемы могут возникнуть при работе с асинхронным кодом, и как их можно решить?
12. Каким образом асинхронные запросы обрабатываются в JavaScript?
13. Какие существуют альтернативы Fetch API для работы с HTTP-запросами?
14. Какие методы шаблонизации вы предпочитаете и почему?
15. Какие аспекты разработки клиент-серверных веб-приложений требуют особого внимания?

Критерии оценки ответов на рубежной аттестации

Регламентом БРС предусмотрено всего 20 баллов за рубежную аттестацию студента. Критерии оценки разработаны, исходя из возможности ответа студентом на 2 вопроса в билете (по 10 баллов).

10 баллов (5+) заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную и дополнительную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, разбирающийся в основных научных концепциях по изучаемой дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала, ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично.

9 баллов (5) заслуживает студент, обнаруживший всестороннее, систематическое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную литературу и знаком с дополнительной литературой, рекомендованной программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению, ответ отличается точностью использованных терминов, материал излагается последовательно и логично.

8 баллов (4+) заслуживает студент, обнаруживший полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

7 баллов (4) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

6 баллов (4-) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, отличавшийся достаточной

активностью на практических (семинарских) и лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы.

5 баллов (3+) заслуживает студент, обнаруживший знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для их самостоятельного устранения.

4 балла (3) заслуживает студент, обнаруживший знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя допущенных погрешностей.

3 балла (3-) заслуживает студент, обнаруживший знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей.

2 балла (2) выставляется студенту, обнаружившему пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не выполнившему самостоятельно предусмотренные программой основные задания, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавшему основные практические, семинарские, лабораторные занятия, допускающему существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.

1 балл — нет ответа (отказ от ответа, представленный ответ полностью не по существу содержащихся в экзаменационном задании вопросов)

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ АКАДЕМИКА М.Д.МИЛЛИОНЩИКОВА**

**Институт цифровой экономики и технологического предпринимательства
Кафедра информационных систем в экономике**

Вопросы к экзамену по дисциплине «Разработка Web-приложений на языке JavaScript»

1. Какова роль JavaScript в веб-разработке? (ОПК-2), (ПК-5)
2. В чем заключаются основы синтаксиса JavaScript?
3. Что такое замыкания в JavaScript и как они используются?
4. Как работают прототипы в JavaScript? (ОПК-2), (ПК-5)
5. Что такое асинхронное программирование и почему оно важно?
6. Какие методы позволяют работать с DOM-элементами?
7. Что такое делегирование событий и в чем его преимущества?
8. Какие типы событий существуют в JavaScript?
9. Что такое промисы и как они помогают в асинхронном программировании?
10. Как использовать `async/await` для управления асинхронными операциями?
11. Как осуществляется работа с API с использованием асинхронных запросов?
12. В чем заключается шаблонизация в веб-разработке? (ОПК-2), (ПК-5)
13. Какие методы используются для создания динамических интерфейсов с помощью шаблонизации?
14. Какие возможности предоставляет Fetch API для обмена данными между клиентом и сервером?
15. Какие практические примеры использования Fetch API вы можете привести?
16. Каким образом JavaScript используется для динамического изменения содержимого веб-страницы?
17. Как работает создание, удаление и модификация DOM-элементов?
18. В чем отличие между синхронным и асинхронным выполнением кода в JavaScript? (ОПК-2), (ПК-5)
19. Как можно организовать обработку событий в JavaScript?
20. Какие основные принципы лежат в основе работы с промисами?

21. Какие методы использования асинхронных запросов к API вы знаете?
22. Как можно оптимизировать работу с клиент-серверными веб-приложениями?
23. Какие методы и технологии используются для упрощения работы с интерфейсом в JavaScript?
24. Какие существуют подходы к шаблонизации в веб-разработке?
25. Какие основные принципы стоит учитывать при разработке динамических интерфейсов? (ОПК-2), (ПК-5)
26. Какие проблемы могут возникнуть при работе с асинхронным кодом, и как их можно решить?
27. Каким образом асинхронные запросы обрабатываются в JavaScript?
28. Какие существуют альтернативы Fetch API для работы с HTTP-запросами?
29. Какие методы шаблонизации вы предпочитаете и почему? (ОПК-2), (ПК-5)
30. Какие аспекты разработки клиент-серверных веб-приложений требуют особого внимания?

Критерии оценки ответов на экзамене

Регламентом БРС предусмотрено всего 20 баллов за экзамен. Критерии оценки разработаны, исходя из возможности ответа студентом на 2 вопроса в билете (по 10 баллов).

10 баллов (5+) заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную и дополнительную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, разбирающийся в основных научных концепциях по изучаемой дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала, ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично.

9 баллов (5) заслуживает студент, обнаруживший всестороннее, систематическое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную литературу и знаком с дополнительной литературой, рекомендованной программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению, ответ отличается точностью использованных терминов, материал излагается последовательно и логично.

8 баллов (4+) заслуживает студент, обнаруживший полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

7 баллов (4) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

6 баллов (4-) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, отличавшийся достаточной активностью на практических (семинарских) и лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы.

5 баллов (3+) заслуживает студент, обнаруживший знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для их самостоятельного устранения.

4 балла (3) заслуживает студент, обнаруживший знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя допущенных погрешностей.

3 балла (3-) заслуживает студент, обнаруживший знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей.

2 балла (2) выставляется студенту, обнаружившему пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не выполнившему самостоятельно предусмотренные программой основные задания, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавшему основные практические, семинарские, лабораторные занятия, допускающему существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.

1 балл — нет ответа (отказ от ответа, представленный ответ полностью не по существу содержащихся в экзаменационном задании вопросов)

Критерии оценки ответов на экзамене

Оценку "отлично" заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка "отлично" выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала.

Оценку "хорошо" заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка "хорошо" выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.

Оценку "удовлетворительно" заслуживает студент, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой. Как правило, оценка "удовлетворительно" выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя.

Оценка "неудовлетворительно" выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка "неудовлетворительно" ставится студентам, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ

Лабораторная работа 1: Разработка базового веб-приложения

Цель:

Научиться создавать основную структуру HTML, добавлять базовое стилевое оформление с использованием CSS, инициализировать JavaScript файл и реализовывать базовый функционал для ввода данных и обработки событий.

Описание работы:

1. Создание основной структуры HTML файла для приложения

- Создайте HTML файл с именем `index.html`.
- Добавьте базовую структуру HTML:

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Базовое веб-приложение</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Добро пожаловать в базовое веб-приложение</h1>
  <form id="dataForm">
    <input type="text" id="inputField" placeholder="Введите данные">
    <button type="button" id="submitButton">Отправить</button>
  </form>
  <div id="outputArea"></div>
  <script src="script.js"></script>
</body>
</html>
```

2. Добавление базового стилевого оформления с использованием CSS

- Создайте CSS файл с именем `styles.css`.
- Добавьте базовые стили для веб-приложения:

```

body {
    font-family: Arial, sans-serif;
    background-color: #f0f0f0;
    margin: 0;
    padding: 20px;
}
h1 {
    color: #333;
}
form {
    margin-top: 20px;
}
input[type="text"] {
    padding: 10px;
    margin-right: 10px;
    border: 1px solid #ccc;
    border-radius: 4px;
}
button {
    padding: 10px 15px;
    background-color: #4CAF50;
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}
button:hover {
    background-color: #45a049;
}
#outputArea {
    margin-top: 20px;
    padding: 10px;
    border: 1px solid #ccc;
    border-radius: 4px;
    background-color: white;
}

```

3. Инициализация JavaScript файла и создание основных переменных

- Создайте JavaScript файл с именем `script.js`.
- Инициализируйте основные переменные и добавьте обработчики событий:


```
document.addEventListener('DOMContentLoaded', () => {
  const inputField = document.getElementById('inputField');
  const submitButton = document.getElementById('submitButton');
  const outputArea = document.getElementById('outputArea');

  submitButton.addEventListener('click', () => {
    const inputValue = inputField.value;
    outputArea.textContent = `Вы ввели: ${inputValue}`;
  });
});
```

Реализация базового функционала: ввод данных, обработка событий

- Убедитесь, что при нажатии на кнопку данные из поля ввода отображаются в элементе outputArea.

Лабораторная работа 2: Динамический интерфейс с использованием DOM

Цель:

Научиться использовать DOM API для динамического изменения содержимого страницы, создавать интерактивные элементы управления и обрабатывать пользовательские действия с помощью JavaScript.

Описание работы:

1. **Использование DOM API для динамического изменения содержимого страницы**
 - Создайте HTML файл с именем index.html.
 - Добавьте базовую структуру HTML:

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Динамический интерфейс</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Динамический интерфейс</h1>
  <div id="dynamicContent"></div>
  <button id="addButton">Добавить элемент</button>
  <script src="script.js"></script>
</body>
</html>
```

2. Создание интерактивных элементов управления: кнопки, формы

- В HTML файл добавьте кнопку и форму для ввода данных:

```
<form id="dataForm">
  <input type="text" id="inputField" placeholder="Введите текст">
  <button type="button" id="submitButton">Отправить</button>
</form>
```

3. Обработка пользовательских действий с помощью JavaScript

- Создайте JavaScript файл с именем `script.js`.
- Добавьте код для динамического изменения содержимого страницы и обработки событий:

```
document.addEventListener('DOMContentLoaded', () => {
  const addButton = document.getElementById('addButton');
  const dynamicContent = document.getElementById('dynamicContent');
  const inputField = document.getElementById('inputField');
  const submitButton = document.getElementById('submitButton');

  // Добавление нового элемента при нажатии на кнопку
  addButton.addEventListener('click', () => {
    const newElement = document.createElement('p');
    newElement.textContent = 'Новый элемент';
    dynamicContent.appendChild(newElement);
  });

  // Обработка отправки формы
  submitButton.addEventListener('click', () => {
    const inputValue = inputField.value;
    const newElement = document.createElement('p');
    newElement.textContent = `Вы ввели: ${inputValue}`;
    dynamicContent.appendChild(newElement);
  });
});
```

Лабораторная работа 3: Работа с серверными запросами

Цель:

Научиться настраивать серверную часть приложения для обработки запросов, использовать Fetch API для отправки асинхронных запросов на сервер и получения данных, а также обрабатывать ответы от сервера и динамически обновлять интерфейс.

Описание работы:

1. Настройка серверной части приложения для обработки запросов

- Установите Node.js и создайте новый проект:

```
mkdir server-app
cd server-app
npm init -y
```

- Установите необходимые зависимости

```
npm install express
```

- Создайте файл server.js и настройте сервер с использованием Express

```
const express = require('express');
const app = express();
const port = 3000;

app.use(express.json());

app.get('/api/data', (req, res) => {
  res.json({ message: 'Привет, мир!' });
});

app.post('/api/data', (req, res) => {
  const { input } = req.body;
  res.json({ message: `Вы отправили: ${input}` });
});

app.listen(port, () => {
  console.log(`Сервер запущен на http://localhost:${port}`);
});
```

2. Использование Fetch API для отправки асинхронных запросов на сервер и получения данных

- Создайте HTML файл `index.html` с основной структурой:

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Работа с серверными запросами</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Работа с серверными запросами</h1>
  <form id="dataForm">
    <input type="text" id="inputField" placeholder="Введите данные">
    <button type="button" id="submitButton">Отправить</button>
  </form>
  <div id="outputArea"></div>
  <script src="script.js"></script>
</body>
</html>
```

- Добавьте JavaScript код для отправки запросов и обработки ответов:

```

document.addEventListener('DOMContentLoaded', () => {
  const inputField = document.getElementById('inputField');
  const submitButton = document.getElementById('submitButton');
  const outputArea = document.getElementById('outputArea');

  // GET запрос для получения данных от сервера
  fetch('/api/data')
    .then(response => response.json())
    .then(data => {
      const newElement = document.createElement('p');
      newElement.textContent = data.message;
      outputArea.appendChild(newElement);
    });

  // Обработка отправки формы и POST запрос на сервер
  submitButton.addEventListener('click', () => {
    const inputValue = inputField.value;

    fetch('/api/data', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json'
      },
      body: JSON.stringify({ input: inputValue })
    })
      .then(response => response.json())
      .then(data => {
        const newElement = document.createElement('p');
        newElement.textContent = data.message;
        outputArea.appendChild(newElement);
      });
  });
});

```

3. Обработка ответов от сервера и динамическое обновление интерфейса

- Убедитесь, что при успешном получении данных от сервера интерфейс обновляется, добавляя новые элементы с полученными данными.

Лабораторная работа 4: Оптимизация и доработка приложения

Цель:

Научиться оптимизировать код JavaScript и CSS для улучшения производительности и читаемости, добавлять дополнительный функционал на основе полученных знаний, а также тестировать и отлаживать приложение.

Описание работы:

1. **Оптимизация кода JavaScript и CSS для улучшения производительности и читаемости**
 - **Оптимизация JavaScript:**
 - Проведите рефакторинг кода для улучшения читаемости.
 - Убедитесь, что переменные и функции имеют понятные и осмысленные имена.
 - Используйте современные возможности языка (например, let, const, стрелочные функции).
 - Минимизируйте количество глобальных переменных.
 - Используйте методы работы с массивами и объектами для упрощения кода.
 - **Оптимизация CSS:**
 - Убедитесь, что стили структурированы и логически разделены.
 - Используйте каскадные свойства и наследование для уменьшения дублирования кода.
 - Минимизируйте использование специфичных селекторов для улучшения производительности.
 - Сократите CSS файл, удалив ненужные стили и комментарии.
2. **Добавление дополнительного функционала на основе полученных знаний**
 - **Пример дополнительного функционала:**
 - Реализуйте фильтрацию данных, отображаемых на странице.
 - Добавьте функционал сортировки данных.
 - Включите валидацию пользовательского ввода с отображением сообщений об ошибках.
 - Добавьте возможность редактирования и удаления элементов, отображаемых на странице.

```
// Пример добавления фильтрации данных
const filterInput = document.getElementById('filterInput');
filterInput.addEventListener('input', () => {
  const filterValue = filterInput.value.toLowerCase();
  const items = document.querySelectorAll('#outputArea p');
  items.forEach(item => {
    if (item.textContent.toLowerCase().includes(filterValue)) {
      item.style.display = '';
    } else {
      item.style.display = 'none';
    }
  });
});
```

3. Тестирование и отладка приложения

- **Тестирование функционала:**

- Напишите тестовые сценарии для проверки основных функций приложения.
- Убедитесь, что все функции работают корректно в различных сценариях использования.

- **Отладка кода:**

- Используйте инструменты разработчика в браузере для поиска и устранения ошибок.
- Убедитесь, что в консоли нет ошибок и предупреждений.
- Проведите стресс-тестирование приложения, чтобы убедиться в его стабильности.

Лабораторная работа 5: Развертывание и документирование

Цель:

Научиться развертывать приложение на хостинге или локальном сервере, создавать документацию для пользователя и разработчика, а также подготавливать проект к презентации и демонстрации.

Описание работы:

1. **Развертывание приложения на хостинге или локальном сервере**
 - **Выбор хостинга или настройка локального сервера:**
 - Выберите платформу для развертывания (например, Heroku, Netlify, Vercel, GitHub Pages).
 - Либо настройте локальный сервер для развертывания (например, с использованием Apache или Nginx).
 - **Подготовка проекта к развертыванию:**
 - Создайте необходимые конфигурационные файлы для выбранной платформы (например, Procfile для Heroku, netlify.toml для Netlify).
 - Убедитесь, что все зависимости указаны в файле package.json (если используется Node.js).
 - **Процесс развертывания:**
 - Загрузите проект на выбранную платформу или локальный сервер.
 - Убедитесь, что приложение работает корректно после развертывания.
 - Проверьте доступность всех функциональных элементов приложения.
2. **Создание документации для пользователя и разработчика**
 - **Документация для пользователя:**
 - Опишите функциональные возможности приложения.
 - Приведите пошаговую инструкцию по использованию приложения.
 - Добавьте скриншоты для иллюстрации основных функций.
 - **Документация для разработчика:**
 - Опишите структуру проекта и назначение основных файлов и директорий.
 - Приведите инструкции по установке и настройке среды разработки.
 - Опишите основные компоненты и модули приложения.
 - Приведите примеры кода для основных функций и методов.
3. **Подготовка проекта к презентации и демонстрации**
 - **Подготовка материалов:**
 - Создайте презентацию, включающую описание цели и задач проекта, обзор функциональности, демонстрацию ключевых функций.
 - Подготовьте скринкасты или видео-демонстрации работы приложения.
 - **Тестирование перед демонстрацией:**
 - Проверьте все функции приложения на предмет корректной работы.
 - Убедитесь, что приложение доступно и работает на разных устройствах и браузерах.

4. Презентация проекта:

- Подготовьте краткое выступление, описывающее процесс разработки, основные проблемы и их решения.
- Демонстрируйте приложение в реальном времени, объясняя каждый шаг и функциональность.

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 1

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какова роль JavaScript в веб-разработке?
2. В чем заключаются основы синтаксиса JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 2

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое замыкания в JavaScript и как они используются?
2. Как работают прототипы в JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 1

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какова роль JavaScript в веб-разработке?
2. В чем заключаются основы синтаксиса JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 2

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое замыкания в JavaScript и как они используются?
2. Как работают прототипы в JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 3

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое асинхронное программирование и почему оно важно?
2. Какие методы позволяют работать с DOM-элементами?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 4

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое делегирование событий и в чем его преимущества?
2. Какие типы событий существуют в JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 5

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое промисы и как они помогают в асинхронном программировании?
2. Как использовать `async/await` для управления асинхронными операциями?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 6

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Как осуществляется работа с API с использованием асинхронных запросов?
2. В чем заключается шаблонизация в веб-разработке?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 7

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какие методы используются для создания динамических интерфейсов с помощью шаблонизации?
2. Какие возможности предоставляет Fetch API для обмена данными между клиентом и сервером?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 8

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какие практические примеры использования Fetch API вы можете привести?
2. Какие методы и инструменты используются для обнаружения и предотвращения утечек конфиденциальных данных из веб-приложений?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 9

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какие меры безопасности следует предпринять при работе с файлами, загружаемыми пользователем в веб-приложениях?
2. Какова роль JavaScript в веб-разработке?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 10

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. В чем заключаются основы синтаксиса JavaScript?
2. Что такое замыкания в JavaScript и как они используются?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 1

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какова роль JavaScript в веб-разработке?
2. В чем заключаются основы синтаксиса JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 2

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое замыкания в JavaScript и как они используются?
2. Как работают прототипы в JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 3

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое асинхронное программирование и почему оно важно?
2. Какие методы позволяют работать с DOM-элементами?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 4

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое делегирование событий и в чем его преимущества?
2. Какие типы событий существуют в JavaScript?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 5

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Что такое промисы и как они помогают в асинхронном программировании?
2. Как использовать `async/await` для управления асинхронными операциями?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 6

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Как осуществляется работа с API с использованием асинхронных запросов?
2. В чем заключается шаблонизация в веб-разработке?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 7

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какие методы используются для создания динамических интерфейсов с помощью шаблонизации?
2. Какие возможности предоставляет Fetch API для обмена данными между клиентом и сервером?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 8

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какие практические примеры использования Fetch API вы можете привести?
2. Какие методы и инструменты используются для обнаружения и предотвращения утечек конфиденциальных данных из веб-приложений?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 9

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. Какие меры безопасности следует предпринять при работе с файлами, загружаемыми пользователем в веб-приложениях?
2. Какова роль JavaScript в веб-разработке?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

Вариант № 10

Дисциплина «Разработка Web-приложений на языке JavaScript»

1. В чем заключаются основы синтаксиса JavaScript?
2. Что такое замыкания в JavaScript и как они используются?

Преподаватель

/М.К. Абдулаев/

Зав. кафедрой

/Л.Р. Магомаева/

КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ К ЭКЗАМЕНУ (8-Й СЕМЕСТР)

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 1

Дисциплина «Разработка Web-приложений на языке JavaScript»

Институт ИЦЭиТП специальность ПИ 5 семестр

1. Какова роль JavaScript в веб-разработке?
2. В чем заключаются основы синтаксиса JavaScript?

УТВЕРЖДЕНО

на заседании кафедры

протокол № ____ от _____

зав. кафедрой

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 2

Дисциплина «Разработка Web-приложений на языке JavaScript»

Институт ИЦЭиТП специальность ПИ 5 семестр

1. Что такое замыкания в JavaScript и как они используются?
2. Как работают прототипы в JavaScript?

УТВЕРЖДЕНО

на заседании кафедры

протокол № ____ от _____

зав. кафедрой

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 3

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Что такое асинхронное программирование и почему оно важно?
2. Какие методы позволяют работать с DOM-элементами?

УТВЕРЖДЕНО

на заседании кафедры

протокол № _____ от _____

зав. кафедрой

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 4

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Что такое делегирование событий и в чем его преимущества?
2. Какие типы событий существуют в JavaScript?

УТВЕРЖДЕНО

на заседании кафедры

протокол № _____ от _____

зав. кафедрой

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 5

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Что такое промисы и как они помогают в асинхронном программировании?
2. Как использовать `async/await` для управления асинхронными операциями?

УТВЕРЖДЕНО

зав. кафедрой

на заседании кафедры

протокол № _____ от _____

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 6

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Как осуществляется работа с API с использованием асинхронных запросов?
2. В чем заключается шаблонизация в веб-разработке?

УТВЕРЖДЕНО

зав. кафедрой

на заседании кафедры

протокол № _____ от _____

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 7

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Какие методы используются для создания динамических интерфейсов с помощью шаблонизации?
2. Какие возможности предоставляет Fetch API для обмена данными между клиентом и сервером?

УТВЕРЖДЕНО

на заседании кафедры

протокол № ____ от _____

зав. кафедрой

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 8

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Какие практические примеры использования Fetch API вы можете привести?
2. Каким образом JavaScript используется для динамического изменения содержимого веб-страницы?

УТВЕРЖДЕНО

на заседании кафедры

протокол № ____ от _____

зав. кафедрой

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 9

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Как работает создание, удаление и модификация DOM-элементов?
2. В чем отличие между синхронным и асинхронным выполнением кода в JavaScript?

УТВЕРЖДЕНО

зав. кафедрой

на заседании кафедры

протокол № ____ от _____

Л.Р. Магомаева

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. акад. М.Д. Миллионщикова**

БИЛЕТ № 10

Дисциплина «Разработка Web-приложений на языке JavaScript»
Институт ИЦЭиТП специальность ПИ 5 семестр

1. Как можно организовать обработку событий в JavaScript?
2. Какие основные принципы лежат в основе работы с промисами?

УТВЕРЖДЕНО

зав. кафедрой

на заседании кафедры

протокол № ____ от _____

Л.Р. Магомаева